To move canvas, hold mouse wheel or spacebar while dragging, or use the hand tool

Yashivardhan Singa

3 - Terminal MTJ

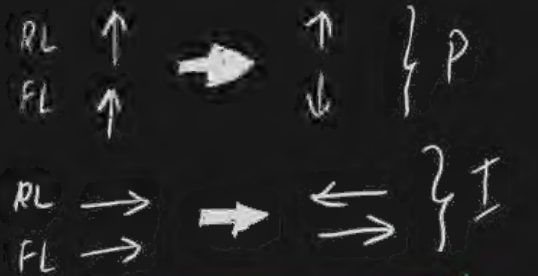
3/9/24

The IDEA

MTJ Classification based on Magnetic Anisotropy

#MTJ
- In-Plane → ⟷ → P2AP / AP2P
- Perendicular → ↕ → P2AP / AP2P

P2AP
Parallel to Antiparalle



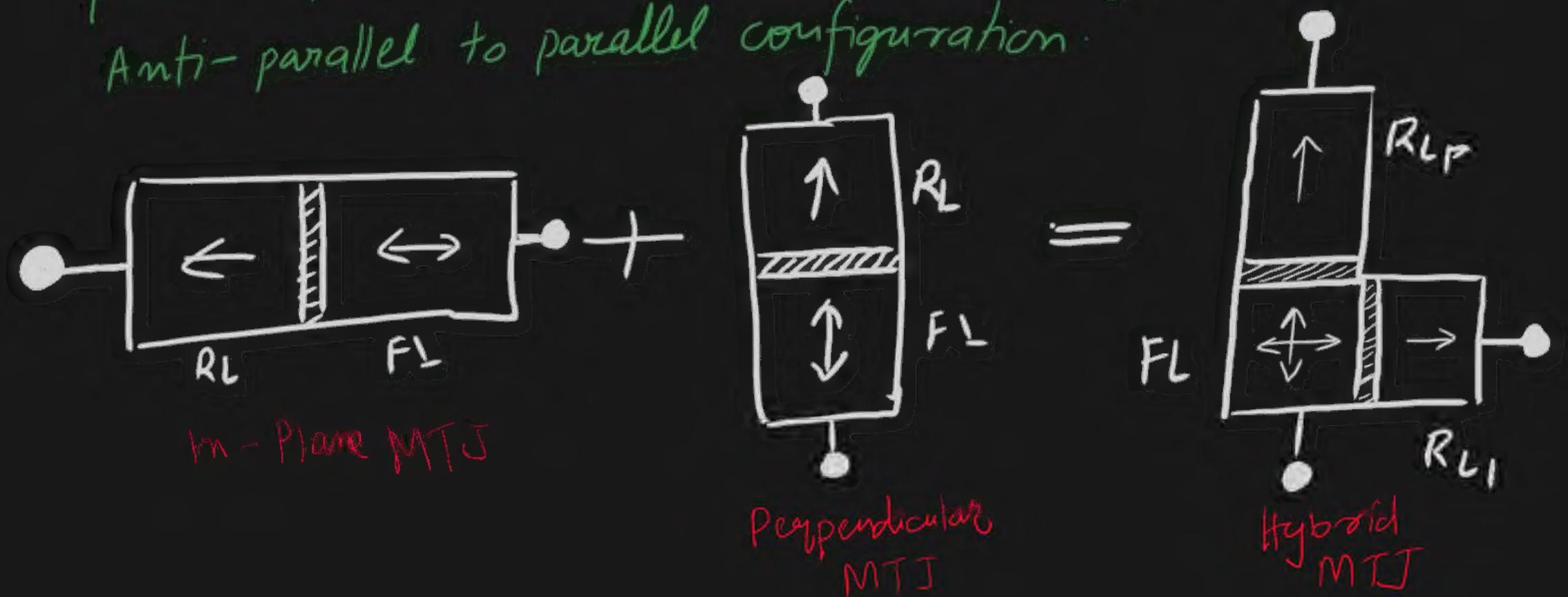
more current needed
↓
less desired

AP 2P
Antiparallel to Parallel

less current needed
↓
desirable condition

I
ntages
'S

The **IDEA** is to combine the pMTJ (perpendicular) and the iMTJ (in-plane) to get a HYBRID MTJ that switches from perpendicular to in-plane so as to only operate in Anti-parallel to parallel configuration.



#

Why?

Optimization of existing CMOS – MTJ circuits

How ?? → [1] Trying and always reversing the MTJ in AP {Anti-parallel} to P {Parallel} configuration.

⇓

CURRENT DEMAND drastically REDUCES

→ [2] Non-volatility
Better Processing Speeds
Thermal Stability

} general advantages of MTJs

HOW

RL
FL

RL
FL

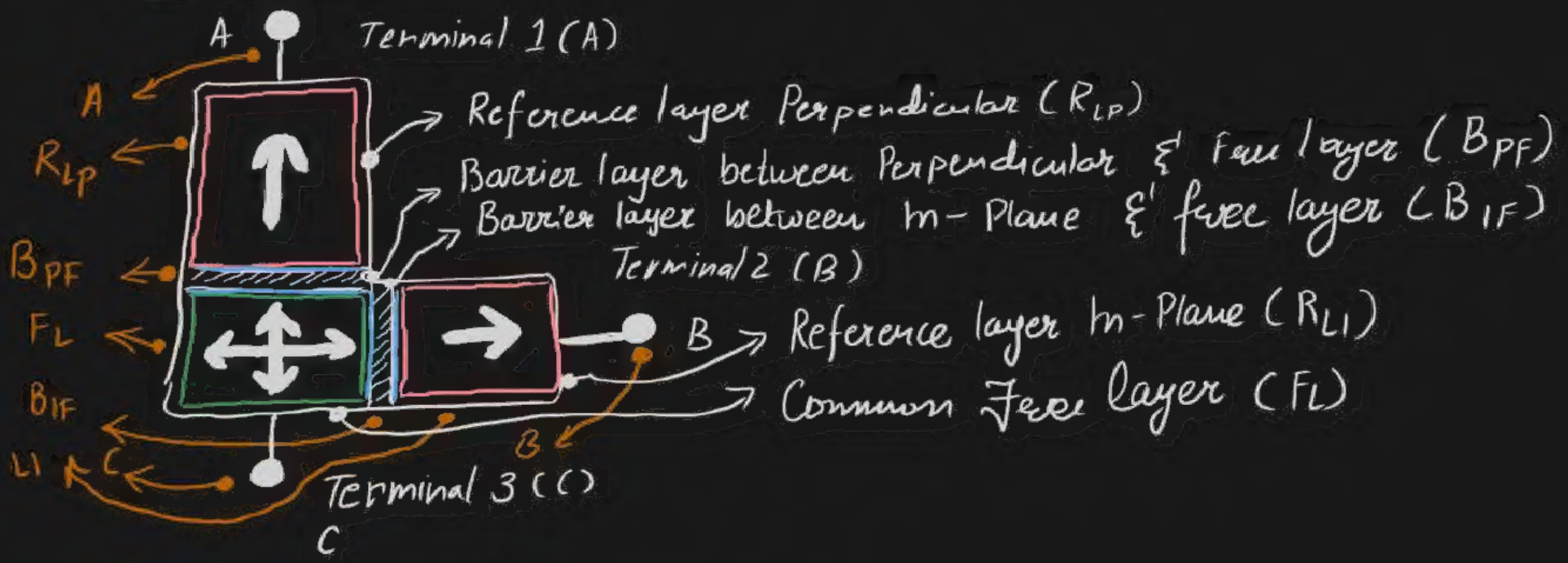
How

Better Processing speeds of MTJs.

Thermal stability

le

Proposed Structure :



→ Barrier layers → SEPARATE

→ Free layer ⟶ COMMON

→ Reference layers → SEPARATE

PROBLEMS :

1) LLGS equation only solves magnetization dynamics for either perpendicular OR in-plane.
∴ incorporating them to work together on a hybrid device needs significant changes

2) lets ignore problem 1 & assume the proposed MTJ somehow works. Even if it switches states from AP to P with respect to pMTJ, in the immediate next switching cycle, the FL state in perpendicular configuration has to be modeled to be Anti-parallel with respect to iMTJ.

3) Corresponding read and write drivers / circuits.

4) Usage & Implementation of this in pre-existing circuits.

Potential Solutions

1) for problem 1,
lets say we divide LLG equat

1)

for problem 1,

lets say we divide LLG equation into 2 components.

$$LLG_{overall} = LLG_{perpendicular} + LLG_{in\text{-}plane}$$

we can introduce a multiplication factor here and call it the "type" parameter. type = 1 → P type = 2 → I

Whenever p [perpendicular] is ON we want the i [in-plane] part to be off, and vice versa.

So, we can use MA in the equation above to selectively activate components.

↳ MA = type ← same values

$$LLG_{overall} = \{[MA]\times[LLG_p]\} + \{[1-MA]\times[LLG_i]\} \rightarrow (1)$$

now, let's say we want the LLG to be solved for the instant where it operates in perpendicular configuration then type = 1,

Substitute in eqn (1):

$$LLG_{overall} = (1 \times \overbrace{LLG_p}^{\text{finite}}) + (\overbrace{(1-1)}^{0} \times LLG_i)$$

$\therefore LLG_{overall} = LLG_p$

Similarly, if it is working in in-plane config:

type = 0,

$$LLG_{overall} = (\overbrace{0}^{0} \times LLG_p) + (\overbrace{(1-0)}^{\text{finite}} \times LLG_i)$$

$\therefore LLG_{overall} = LLG_i$

2] 2.1) Investigation into easy axis based switching

↳ ongoing

2.2) mapping P states to I states by developing some algorithm

Problem 3 & 4

↳ ideal to look into post-model development as they are crucial implementation concerns.

Thank you!